



AF

**THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of:

Ralf Uwe Krauklis

Serial No. 09/464,021

Filed: December 15, 1999

For: System and Method for Managing a  
Scalable List of Items for  
Display

§ Group Art Unit: 2174

§

§ Examiner: Luu, Sy D

§

§ Atty. Dkt. No.: 5181-53800

§

P4599

§

§

§

§

§

§

§

§

§

§

§

§

**CERTIFICATE OF MAILING**

37 C.F.R. § 1.8

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date indicated below:

B. Noël Kivlin

Name of Registered Representative

10-04-04

Date

Signature

**APPEAL BRIEF**

**Box AF**

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir/Madam:

Further to the Notice of Appeal filed August 3, 2004, Appellant presents this Appeal Brief. Appellant respectfully requests that this appeal be considered by the Board of Patent Appeals and Interferences.

10/08/2004 HUUONG1 00000027 501505 09464021

01 FC:1402 340.00 DA

## **I. REAL PARTY IN INTEREST**

The subject application is owned by Sun Microsystems, Inc., a corporation organized and existing under and by virtue of the laws of the State of Delaware, and having its principal place of business at 4150 Network Circle, Santa Clara, CA 95054, as evidenced by the assignment recorded at Reel 010704, Frame 0602.

## **II. RELATED APPEALS AND INTERFERENCES**

This appeal is not related to any other appeals.

## **III. STATUS OF CLAIMS**

Claims 1 – 19 are pending and rejected. The rejection of claims 1 – 19 is being appealed. A copy of claims 1 – 19 is included in the Claims Appendix hereto.

## **IV. STATUS OF AMENDMEMNTS**

No amendments to the claims have been filed subsequent to the final rejection.

## **V. SUMMARY OF CLAIMED SUBJECT MATTER**

Various embodiments, such as a system (e.g. claim 1, claim 13), method (e.g., claim 7) and memory medium (claim 18), may provide for managing a scalable list of items for display in a display device of a small footprint device. A client program running in a small footprint device may instantiate a list container object. The list container object may then add list item data objects to the list container object. *See* specification, page 4, lines 3 – 9.

The number of list items data objects that can be displayed by the list container object at a given time depends on the size of the list, i.e., the number of “rows” the list container object is configured with. For example, if the list container object maintains ten

list item data objects but is only configured to have four rows, then only a subset of the list item data objects may be displayed. The list container object may maintain a “start index” that specifies which of the list item data objects is currently displayed in the first row of the visible list. *See* specification, page 11, lines 1 –13.

The list container object may interface with a number of item renderer objects. Each list item renderer object may correspond to a row in the displayed list, and may implement code for appropriately displaying a list item data object. Specifically, each list item renderer object is operable to display a list item data object in the user interface, as appropriate for the particular type of that list item data object. For example, for list item data objects representing e-mail messages, the list item renderer objects may retrieve information from the message objects, such as the sender’s name, the message subject, etc., and may then display this information in the user interface. *See* specification, page 11, lines 11 – 16 and page 12, lines 2 – 7.

Appellant’s claimed invention may provide a general framework in which any of various types of objects may be displayed. *See* specification, page 4, lines 17 – 20.

## **VI. GROUND OF REJECTION TO BE REVIEWED ON APPEAL**

I. Claims 1 – 19 are rejected under 35 U.S.C. § 102(e) as being anticipated by Pogue (PalmPilot: The Ultimate Guide, hereinafter “Pogue”).

## **VII. ARGUMENT**

### **First Ground of Rejection:**

Claims 1 – 19 are finally rejected under 35 U.S.C. § 102(e) as being anticipated by Pogue (PalmPilot: The Ultimate Guide, hereinafter “Pogue”). Appellant traverses this rejection for the following reasons. Different groups of claims are addressed under their

respective subheadings.

### **Claims 1, 7, 13, 17, and 18**

Claim 1 stands rejected under 35 U.S.C. § 102(e) are unpatentable over Pogue. Appellant asserts that the Examiner has not established a *prima facie* case that Pogue teaches the system described in Appellant's claim 1.

In rejecting Claim 1, the Examiner asserts on page 5 of the Final Office Action dated May 18, 2004, that "a container object is an object that contains/holds objects such as item data objects," and that "according to the language of the claims, the list container object could be interpreted as a directory that contains child objects or item data objects."

However, Appellant disagrees with the Examiner's assertions. A directory, as is commonly understood in the art, is a data structure, and as such is **not** "executable to specify a corresponding list item data object for each of a plurality of list item renderer objects," as recited in Appellant's claim 1.

Likewise, Appellant disagrees with the Examiner's assertion that "each item data object has a corresponding renderer object, which could be interpretable as internal data that is associated with an item data object, and being used for the system to render/display the item data object." Internal data is **not** executable, and as such cannot be characterized as a "list item renderer object" "executable to display the list item data object in the display device of the small footprint device," as recited in claim 1.

Furthermore, even if, *arguendo*, the "internal data" is assumed to be associated with an item data object and used to render/display the item data object, Appellant can find no teaching or suggestion in Pogue wherein such action is taken "in response to said list container object specifying a corresponding list item data object for each list item renderer object," as recited in claim 1.

For these reasons, Appellant respectfully submits that claim 1 patentably distinguishes over Pogue.

Claims 7, 13, and 18 recite features similar to those of claim 1 as discussed above and are thus also believed to distinguish over the cited reference for at least the same reasons. Claim 17 is dependent upon claim 13, and is thus also believed to distinguish over the cited reference.

### **Claims 2, 8, 14, and 19**

All the arguments given above in regard to claim 1 apply to claim 2.

Furthermore, even if, *arguendo*, Pogue's directory is assumed to be a list container object, Appellant can find no teaching or suggestion in Pogue that the directory **"is executable to instantiate the plurality of list item renderer objects,"** as recited in Appellant's claim 2.

Accordingly, Pogue does not teach or suggest the features identified above with respect to claim 2.

Claims 8, 14 and 19 recite similar features to claim 2 and are thus also believed patentable.

### **Claims 3, 9, and 15**

All the arguments given above in regard to claim 2 apply to claim 3.

Furthermore, Pogue does not teach a client program, **"wherein the client program is executable to provide the list container object with information specifying a list item render object class,"** as recited in Appellant's claim 3, nor does Pogue teach a system

**“wherein said list container object instantiating a plurality of list item renderer objects comprises the list container object instantiating a plurality of objects of the list item renderer object class specified by the program,”** as recited in Appellant’s claim 3.

Accordingly, Pogue does not teach or suggest the features identified above with respect to claim 3. Claims 9 and 15 recite similar features to claim 3 and are thus also believed to distinguish over the cited reference for at least the same reasons.

#### **Claims 4, 10, and 16**

All the arguments given above in regard to claim 3 apply to claim 4.

Furthermore, Appellant can find no teaching or suggestion of **“an item renderer interface,”** or a **“‘set data’ method”** in Pogue. Furthermore, Appellant can find no teaching or suggestion of a system wherein **“specifying a corresponding list item data object for each list item renderer object comprises the list container object passing the corresponding list item data object to the ‘set data’ method for each list item renderer object,”** as recited in Appellant’s claim 4.

Accordingly, Pogue does not teach or suggest the features identified above with respect to claim 4. Claims 10 and 16 recite similar features to claim 4 and are thus also believed to distinguish over the cited reference for at least the same reasons.

#### **Claims 5 and 11**

All the arguments given above in regard to claim 1 apply to claim 5.

Appellant disagrees with the Examiner’s assertion on page 3 of the Final Office Action dated May 18, 2004, that **“inherently, the list container object must update the pointer index to the first list item data object shown on the display when a scrolling operation takes place in order to manage the display of list item data objects properly.”**

Pogue shows a Palm Pilot with a plurality of application icons and a scroll bar on page 4. However, Appellant can find no teaching or suggestion in Pogue that the application icons are contained in a list container object, or that the list container object **“maintains a start index specifying the first list item data object currently being displayed,”** as recited in Appellant’s claim 5.

Furthermore, Appellant can find no teaching or suggestion in Pogue that **“in response to user interaction, the list container object is executable to update the start index and specify an updated list item data object corresponding to each of the plurality of list item renderer objects,”** or that **“in response to said list container object specifying an updated list item data object corresponding to each list item renderer object, each list item renderer object is executable to display the updated list item data object in the display device of the small footprint device,”** as recited in Appellant’s claim 5.

Accordingly, Pogue does not teach or suggest the features identified above with respect to claim 5. Claim 12 recites similar features to claim 5 and is thus also believed to distinguish over the cited reference for at least the same reasons.

### **Claims 6 and 12**

All the arguments given above in regard to claim 1 apply to claim 6.

Appellant can find no teaching or suggestion in Pogue that **“each of the plurality of list item renderer objects corresponds to a list row displayed in the display device of the small footprint device,”** or that **“each list item renderer object displaying the list item data object in the display device of the small footprint device comprises each list item renderer object displaying the list item data object in the list row corresponding to the list item renderer object,”** as recited in Appellant’s claim 6.

Accordingly, Pogue does not teach or suggest the features identified above with respect to claim 6. Claim 12 recites similar features to claim 6 and is thus also believed to distinguish over the cited reference for at least the same reasons.



### VIII. CONCLUSION

For the foregoing reasons, it is submitted that the Examiner's rejection of claims 1 – 19 was erroneous, and reversal of the Examiner's decision is respectfully requested.

This Appeal Brief is submitted in triplicate along with the following items:

☒ Return Receipt Postcard

☒ Deposit Account Fee Authorization form for the \$320.00 appeal brief fee.

Respectfully submitted,



B. Noël Kivlin  
Reg. No. 33,929  
Attorney for Appellant

Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C.  
P.O. Box 398  
Austin, TX 78767-0398  
(512) 853-8800  
Date: 10-04-04



## IX. APPENDIX A

The claims on appeal are as follows.

1. (Original) A system for managing a scalable list of items for display, the system comprising:

a small footprint device including a display device and a CPU coupled to a memory;

a client program comprised in the memory of the small footprint device, wherein the client program is executable to instantiate a list container object and add list item data objects to the list container object;

wherein the list container object is executable to specify a corresponding list item data object for each of a plurality of list item renderer objects;

wherein, in response to said list container object specifying a corresponding list item data object for each list item renderer object, each list item renderer object is executable to display the list item data object in the display device of the small footprint device.

2. (Original) The system of claim 1,

wherein the list container object is executable to instantiate the plurality of list item renderer objects.

3. (Original) The system of claim 2,

wherein the client program is executable to provide the list container object with information specifying a list item renderer object class;

wherein said list container object instantiating a plurality of list item renderer objects comprises the list container object instantiating a plurality of objects of the list item renderer object class specified by the client program.

4. (Original) The system of claim 3,

wherein the list item renderer object class implements an item renderer interface;

wherein the item renderer interface includes a "set data" method to set the list item data object corresponding to a list item renderer object;

wherein said list container object specifying a corresponding list item data object for each list item renderer object comprises the list container object passing the corresponding list item data object to the “set data” method for each list item renderer object.

5. (Original) The system of claim 1,

wherein the list container object maintains a start index specifying the first list item data object currently being displayed;

wherein, in response to user interaction, the list container object is executable to update the start index and specify an updated list item data object corresponding to each of the plurality of list item renderer objects;

wherein, in response to said list container object specifying an updated list item data object corresponding to each list item renderer object, each list item renderer object is executable to display the updated list item data object in the display device of the small footprint device.

6. (Original) The system of claim 1,

wherein each of the plurality of list item renderer objects corresponds to a list row displayed in the display device of the small footprint device;

wherein said each list item renderer object displaying the list item data object in the display device of the small footprint device comprises each list item renderer object displaying the list item data object in the list row corresponding to the list item renderer object.

7. (Original) A method for managing a scalable list of items for display in a display device of a small footprint device, the method comprising:

a client program comprised in the memory of the small footprint device including a CPU and memory instantiating a list container object and adding list item data objects to the list container object;

the list container object specifying a corresponding list item data object for each of a plurality of list item renderer objects;

in response to said list container object specifying a corresponding list item data object for each list item renderer object, each list item renderer object displaying the list item data object in the display device of the small footprint device.

8. (Original) The method of claim 7, further comprising:  
the list container object instantiating the plurality of list item renderer objects.
9. (Original) The method of claim 8, further comprising:  
the client program providing the list container object with information specifying a list item renderer object class;  
wherein said list container object instantiating a plurality of list item renderer objects comprises the list container object instantiating a plurality of objects of the list item renderer object class specified by the client program.
10. (Original) The method of claim 9,  
wherein the list item renderer object class implements an item renderer interface;  
wherein the item renderer interface includes a “set data” method to set the list item data object corresponding to a list item renderer object;  
wherein said list container object specifying a corresponding list item data object for each list item renderer object comprises the list container object passing the corresponding list item data object to the “set data” method for each list item renderer object.
11. (Original) The method of claim 7, further comprising:  
the list container object maintaining a start index specifying the first list item data object currently being displayed;  
in response to user interaction, the list container object updating the start index and specifying an updated list item data object corresponding to each of the plurality of list item renderer objects;

in response to said list container object specifying an updated list item data object corresponding to each list item renderer object, each list item renderer object displaying the updated list item data object in the display device of the small footprint device.

12. (Original) The method of claim 7,

wherein each of the plurality of list item renderer objects corresponds to a list row displayed in the display device of the small footprint device;

wherein said each list item renderer object displaying the list item data object in the display device of the small footprint device comprises each list item renderer object displaying the list item data object in the list row corresponding to the list item renderer object.

13. (Original) A system comprising:

a central processing unit (CPU);

memory coupled to the CPU;

a display device;

a client program comprised in the memory, wherein the client program is executable to instantiate a list container object and add list item data objects to the list container object;

wherein the list container object is executable to specify a corresponding list item data object for each of a plurality of list item renderer objects;

wherein, in response to said list container object specifying a corresponding list item data object for each list item renderer object, each list item renderer object is executable to display the list item data object in the display device.

14. (Original) The system of claim 13,

wherein the list container object is executable to instantiate the plurality of list item renderer objects.

15. (Original) The system of claim 14,

wherein the client program is executable to provide the list container object with information specifying a list item renderer object class;

wherein said list container object instantiating a plurality of list item renderer objects comprises the list container object instantiating a plurality of objects of the list item renderer object class specified by the client program.

16. (Original) The system of claim 15,

wherein the list item renderer object class implements an item renderer interface;

wherein the item renderer interface includes a “set data” method to set the list item data object corresponding to a list item renderer object;

wherein said list container object specifying a corresponding list item data object for each list item renderer object comprises the list container object passing the corresponding list item data object to the “set data” method for each list item renderer object.

17. (Original) The system of claim 13,

wherein the CPU, memory, and display device are included within a small footprint device.

18. (Original) A memory medium comprising program instructions which implement:

a list container object specifying a corresponding list item data object for each of a plurality of list item renderer objects;

in response to said list container object specifying a corresponding list item data object for each list item renderer object, each list item renderer object displaying the list item data object in a display device of a system.

19. (Original) The memory medium of claim 18, further comprising program instructions which implement:

the list container object instantiating the plurality of list item renderer objects.

**X. EVIDENCE APPENDIX**

No evidence submitted under 37 CFR §§ 1.130, 1.131 or 1.132 or otherwise entered by the Examiner is relied upon in this appeal.

**XI. RELATED PROCEEDINGS APPENDIX**

There are no related proceedings.





IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application No.: 09/464,021  
Filed: December 15, 1999  
Inventor(s):  
Ralf Uwe Krauklis

Title: System and Method for  
Managing a Scalable List of  
Items for Display

§ Examiner: Sy D. Luu  
§ Group/Art Unit: 2174  
§ Atty. Dkt. No: 5181-53800  
§  
§  
§  
§  
§  
§  
§  
§  
§  
§

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date indicated below.

B. Noël Kivlin

Printed Name

  
Signature

October 4, 2004

Date

FEE AUTHORIZATION

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

The Commissioner is hereby authorized to charge the following fee to Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C. Deposit Account Number 50-1505/5181-53800:

Fee: Appeal Brief  
Amount: \$340.00  
Attorney Docket No.: 5181-53800

The Commissioner is also authorized to charge any extension fee or other fees which may be necessary to the same account number.

Respectfully submitted,



B. Noël Kivlin  
Reg. No. 33,929  
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel, P.C.  
P.O. Box 398  
Austin, Texas 78767-0398  
Phone: (512) 853-8800  
Date: 10/9/04